

DAILY ONLINE ACTIVITIES SUMMARY

Date:	18-05-2020	Name:	Shetty Sonali Sanjeeva
Sem & Sec	8 th , B	USN:	4AL16CS123
Online Test Summary			
Subject	System Modelling and Simulation		
Max. Marks	60	Score	45
Certification Course Summary			
Course	MACHINE LEARNING ONRAMP BY MATHWORKS		
Certificate Provider	ICT academy	Duration	2 Hrs
Coding Challenges			
Problem Statement: Finding frequency of each character in a string and to print the even and odd for series			
Status: Solved			
Uploaded the report in Github		Yes	
If yes Repository name		SONALISHETTY	
Uploaded the report in slack		Yes	

Online Test Details: (Attach the snapshot and briefly write the report for the same)

Certification Course Details: (Attach the snapshot and briefly write the report for the same)

Coding Challenges Details: (Attach the snapshot and briefly write the report for the same)

1) Online Test Details:

The screenshot shows a web browser window with the URL `techgig.com/challenge/SMS?utm_source=Mailer&utm_medium=TG_batch&utm_campaign=Act_contestskilltestresult_2020-05-18&email=shettysonali1...`. The page features a banner with a play button icon and the text "Challenge Over by TechGig SMS_I_IA Enhanced Clone At 2020-05-18 10:47:27". Below the banner, there are two main sections: "MCQ" and "Summary".

MCQ Section:

- Header: MCQ
- Text: Your Highest Score 45 Max Score 60
- Button: Start Test

Summary Section:

Skills	Problem Solving Skills
Ends On	18 May

Details Section:

- Tab: Details (selected)
- Section: SMS FIRST IA RE ASSESSMENT
- Section: Rules
 - 1. Any participant can attempt the assessment only 1 times, Only your best score counts!!
 - 2. This test has negative marking as per the following rules -
 - MCQ

The Windows taskbar at the bottom shows the search bar, task view button, and several application icons (Excel, Firefox, Chrome, Edge, Teams, Word, PowerPoint). The system clock indicates 10:42 AM on 20-May-20.

2) Certification Course Details:



Course Completion Certificate

Shetty Sonali

has successfully completed 100% of the self-paced training course

Machine Learning Onramp

A handwritten signature in black ink, reading 'Craig Santos', written over a horizontal line.

DIRECTOR, TRAINING SERVICES

07 May 2020

3) Coding Challenges:

1. Write down a java program to print even and odd numbers series respectively from two threads: t1 and t2 synchronizing on a shared object
Let t1 print message “ping—>” and t2 print message “,—pong”.
Take as command line arguments, the following inputs to the program:
Sleep Interval for thread t1
2. Write down a java program to print even and odd numbers series respectively from two threads: t1 and t2 synchronizing on a shared object

Let t1 print message “ping — >” and t2 print message “,—pong”.

Take as command line arguments, the following inputs to the program:

Sleep Interval for thread t1

Sleep Interval for thread t2

Message per cycle

No of cycles

PROGRAM1

```
import java.util.Scanner;
public class StringOperators
{
    public static void main(String args[])

        int i;
        String str;

        int counter[] = new int[256];
        Scanner in = new Scanner(System.in);

        System.out.print("Enter a String : ");
        str=in.nextLine();

        for (i = 0; i < str.length(); i++) {
            counter[(int) str.charAt(i)]++;
        }
        // Print Frequency of characters
        for (i = 0; i < 256; i++) {
            if (counter[i] != 0) {
                System.out.println((char) i + ":-" + counter[i] + " times");
            }
        }
    }
}
```

PROGRAM 2

```

public class PingPong extends Thread {
static StringBuilder object = new StringBuilder("");

public static void main(String[] args) throws InterruptedException {

Thread t1 = new PingPong();
Thread t2 = new PingPong();

t1.setName("\nping");
t2.setName(" pong");

t1.start();
t2.start();
}

@Override
public void run() {
    working();
}

void working() {
while (true) {
    synchronized (object) {
try {
        System.out.print(Thread.currentThread().getName());
        object.notify();
        object.wait();
    } catch (InterruptedException e) {
        e.printStackTrace();
    }
}
}
}
}
}
}

```